

Publication number: JP10133618

Publication date: 1998-05-22

Inventor: KANO MAMORU; KUROKAWA MASUYOSHI;
NAKAMURA KENICHIRO; IWASE SEIICHIRO

Applicant: SONY CORP

Classification:

- international: *G06T3/40; G09G3/20; G09G3/36; G09G5/00;
G09G5/36; G09G5/391; H04N7/01; G06T3/40;
G09G3/20; G09G3/36; G09G5/00; G09G5/36;
H04N7/01; (IPC1-7): G09G3/20; G06T3/40; G09G3/36;
G09G5/00; G09G5/36; H04N7/01*

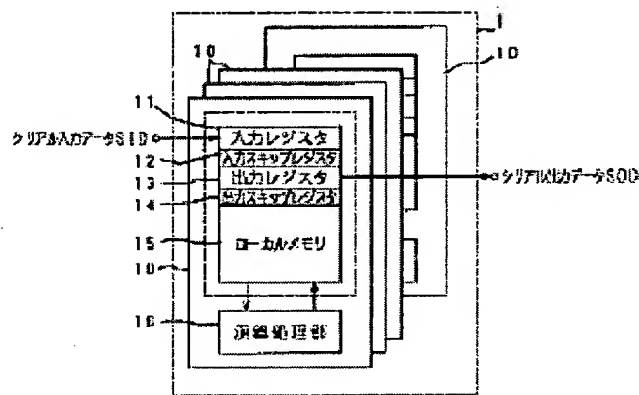
- european:

Application number: JP19960287104 19961029

Priority number(s): JP19960287104 19961029

Abstract of JP10133618

PROBLEM TO BE SOLVED: To realize the number-of-pixel conversion and number-of-scanning line conversion of an arbitrary rate and to deal with change of bit accuracy, addition of new format specification or the like after designing. **SOLUTION:** Each element processor 10 has a local memory 15 storing pixel data and a phase information number, an input register 11 transferring input pixel data to a local memory 15, an input skip register 12 and an output skip register 14 storing pixel skip information, an operation processing section 16 performing a prescribed filter operation based on a phase information number, and an output register 13 storing input pixel data taken out from the memory 15 or pixel data after operation, the processors 10 store discretely input pixel data in the input register 11 based on pixel skip information at the time of enlarging the number of pixels, inversely, store discretely pixel data from the memory 15 in the output register 14 based on pixel skip information in reducing the number of pixels.



Data supplied from the **esp@cenet** database - Worldwide

THIS PAGE BLANK (USPTO)

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-133618

(43) 公開日 平成10年(1998) 5月22日

(51) Int.Cl.⁶

識別記号

F I

G 0 9 G 3/20

G 0 9 G 3/20

U

G 0 6 T 3/40

3/36

G 0 9 G 3/36

5/00

5 2 0 V

5/00

5 2 0

5/36

5 2 0 C

5/36

5 2 0

H 0 4 N 7/01

C

審査請求 未請求 請求項の数 8 O L (全 19 頁) 最終頁に続く

(21) 出願番号

特願平8-287104

(22) 出願日

平成 8 年 (1996) 10月29日

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川 6 丁目 7 番35号

(72) 発明者 加納 護

東京都品川区北品川 6 丁目 7 番35号 ソニ

ー株式会社内

(72) 発明者 黒川 益義

東京都品川区北品川 6 丁目 7 番35号 ソニ

ー株式会社内

(72) 発明者 中村 憲一郎

東京都品川区北品川 6 丁目 7 番35号 ソニ

ー株式会社内

(74) 代理人 弁理士 小池 晃 (外 2 名)

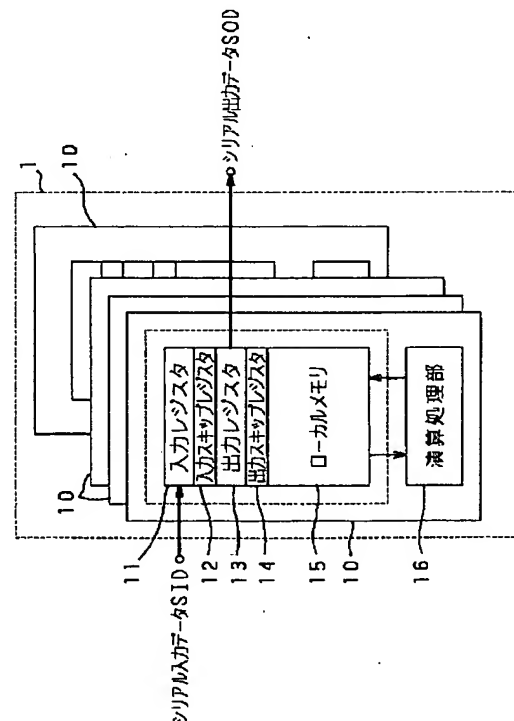
最終頁に続く

(54) 【発明の名称】 画像信号処理装置

(57) 【要約】

【課題】 任意比率の画素数変換や走査線数変換を実現可能にし、設計後のビット精度の変更や新たなフォーマット仕様の追加等に対応できるようにする。

【解決手段】 各要素プロセッサ 10 は、画素データと位相情報番号を保存するローカルメモリ 15 と、入力画素データをローカルメモリ 15 に転送する入力レジスタ 11 と、画素スキップ情報を格納する入力スキップレジスタ 12 及び出力スキップレジスタ 14 と、位相情報番号に基づいて所定のフィルタ演算を行う演算処理部 16 と、ローカルメモリ 15 から取り出された入力画素データ又は演算後の画素データを格納する出力レジスタ 13 とを有し、画素数拡大時には画素スキップ情報に基づいて入力レジスタ 11 に入力画素データを離散的に格納し、逆に画素数縮小時には画素スキップ情報に基づいて出力レジスタ 14 にローカルメモリ 15 からの画素データを離散的に格納する。



【特許請求の範囲】

【請求項 1】 デジタル化された 2 次元画像の 1 次元方向の各画素に対応して配置すると共に上記 1 次元方向の各画素データが時系列に順次入力する複数の要素プロセッサと、各要素プロセッサを共通に制御するための制御手段とを備える画像信号処理装置であって、上記各要素プロセッサは、画素データを一時的に保存する一時保存手段と、入力画素データを格納して上記一時保存手段に転送する入力画素データ格納手段と、画素の属性を表す画素属性情報を格納する画素属性情報格納手段と、画素データをスキップさせる画素スキップ情報を格納する画素スキップ情報格納手段と、上記画素属性情報に基づいて上記入力画素データ又は近傍の要素プロセッサの画素データを用いた所定の演算を行い得られた画素データを上記一時保存手段に保存する算術演算手段と、上記一時保存手段から取り出された画素データを格納して出力する出力画素データ格納手段とを有してなることを特徴とする画像信号処理装置。

【請求項 2】 上記各要素プロセッサの入力画素データ格納手段は、上記画素スキップ情報格納手段が格納する画素スキップ情報に基づいて、上記入力画素データを離散的或いは連続的に格納することを特徴とする請求項 1 記載の画像信号処理装置。

【請求項 3】 上記各要素プロセッサの出力画素データ格納手段は、上記画素スキップ情報格納手段が格納する画素スキップ情報に基づいて、上記一時記憶手段からの画素データを離散的或いは連続的に格納することを特徴とする請求項 1 記載の画像信号処理装置。

【請求項 4】 上記制御手段は、上記出力画素データ格納手段から出力する画素データのレートを、上記入力画素データ格納手段に入力する画素データのレートとは独立に制御することを特徴とする請求項 1 記載の画像信号処理装置。

【請求項 5】 上記各要素プロセッサは、上記画素属性情報を生成する画素属性情報生成手段を備えることを特徴とする請求項 1 記載の画像信号処理装置。

【請求項 6】 上記画素スキップ情報を生成する画素スキップ情報生成手段と、上記画素属性情報を生成する画素属性情報生成手段と、上記入力画素データと、上記画素スキップ情報及び上記画素属性情報とを、切り換えて上記各要素プロセッサに入力するための入力切換手段と、上記入力切換手段を所定のタイミングで切換制御する切換制御手段とを備えることを特徴とする請求項 1 記載の画像信号処理装置。

【請求項 7】 上記画素スキップ情報生成手段及び上記画素属性情報生成手段は順序演算回路からなることを特徴とする請求項 6 記載の画像信号処理装置。

【請求項 8】 上記切換制御手段は、ブランキング期間の任意の 1 走査線時間に、上記入力切換手段が上記画素

スキップ情報及び上記画素属性情報を上記各要素プロセッサに切り換え入力するための切換タイミング信号を生成することを特徴とする請求項 6 記載の画像信号処理装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、画素数変換処理や走査線変換等の画像信号処理を行う画像信号処理装置に関する。

【0002】

【従来の技術】 近年になって、半導体技術、半導体の処理スピード性能の向上により、映像信号のデジタル信号処理が行われるようになってきた。また、最近では画像表示装置も従来のブラウン管に代わり、LCD (Liquid Crystal Display: 液晶ディスプレイ) 表示装置やプラズマディスプレイ装置等に代表される固定画素表示装置が広く普及しつつある。

【0003】 また、最近は、いわゆるNTSC (National Television System Committee) 信号、PAL (Phase Alternation by Line) 信号などの標準テレビジョン放送方式のみならず、HDTV (High Definition Television) 信号や、VGA (Video Graphics Array) 信号、SVGA (Super VGA) 信号、XVGA (extended VGA) 信号など様々なフォーマットの信号を表示できることが求められている。

【0004】 これら様々なフォーマットでは、それぞれ扱う画素数がまちまちである。このようなそれぞれ画素数が異なる各種のフォーマットの映像信号を表示する場合、上記ブラウン管等のアナログ表示デバイスであれば、1 走査線時間当たりの画素数に応じて電子ビームの偏向速度を変えてやれば済む。

【0005】 しかし、上記固定画素表示装置においては、扱える画素数が固定しているため、上述のブラウン管の場合のような従来のアナログ技術は使えない。そのため、これら様々なフォーマットの信号を上述のような固定画素表示装置に対して表示させるためには、デジタル信号処理による任意の画素数変換、或いは走査線数変換が不可欠である。

【0006】 上述の画素数変換処理について、以下にその概要を説明する。

【0007】 画素数変換処理とは、1 走査線期間において入力画素数に対して出力画素数を所望の画素数に増減する処理であり、例えば入出力のサンプリング周波数が同じであるとした場合に、画素数を増加させたならば入力画像の拡大処理（拡大画素数変換処理）となり、逆に画素数を減少させたならば入力画像の処理（縮小画素数変換処理）となる。別の言い方として、画素数ではなく、入出力の画素と画素のサンプリングという点でとらえれば、元々あるサンプリング点のデータから、元々のサンプリング位置とは異なる点のデータを作り出すこと

になり、この異なる点のデータを、入力された画素データから補間により補間画素を生成することに相当する。

【0008】この補間方法には様々な方法があり、ここではその一例として後述するキュービック補間関数を用いた補間法について説明する。

【0009】先ず、例えば入力画素2個に対して出力画素3個を作り出すような2:3拡大画素数変換の原理について説明する。

【0010】図11には上記2:3拡大画素数変換の原理を説明するための図を示す。なお、この図11では、各入力画素の値をそれぞれ R_{i-1} , R_i , R_{i+1} , R_{i+2} , R_{i+3} , ...とし、各出力画素の値をそれぞれ Q_j , Q_{j+1} , Q_{j+2} , Q_{j+3} , ...として表している。また、図11の中の P_1 , P_2 , P_3 , P_1 , ...は、入力画素と出力画素の位相のずれ(位相情報)を表している。

【0011】ここで、上記2:3拡大画素数変換においては、この図11のように入力画素2個に対して出力画

$$\begin{aligned} \text{Cub}(x) &= |x|^3 - 2|x|^2 + 1 & (|x| \leq 1 \text{ の時}) \\ \text{Cub}(x) &= -|x|^3 + 5|x|^2 - 8|x| + 4 & (1 < |x| \leq 2 \text{ の時}) \\ \text{Cub}(x) &= 0 & (2 < |x| \text{ の時}) \\ &\dots (1) \end{aligned}$$

拡大画素数変換の場合、各出力画素の補間値は、入力画素のサンプリング値とキュービック関数との畳み込み演算

$$\begin{aligned} Q_j &= \text{Cub}(x_{11}) * R_{i-1} + \text{Cub}(x_{12}) * R_i + \text{Cub}(x_{13}) * R_{i+1} + \text{Cub}(x_{14}) * R_{i+2} \\ Q_{j+1} &= \text{Cub}(x_{21}) * R_{i-1} + \text{Cub}(x_{22}) * R_i + \text{Cub}(x_{23}) * R_{i+1} + \text{Cub}(x_{24}) * R_{i+2} \\ Q_{j+2} &= \text{Cub}(x_{31}) * R_i + \text{Cub}(x_{32}) * R_{i+1} + \text{Cub}(x_{33}) * R_{i+2} + \text{Cub}(x_{34}) * R_{i+3} \\ &\dots (2) \end{aligned}$$

この式(2)の各係数 $\text{Cub}(x)$ は前記キュービック補間関数から計算される値であり、これは、求めるべき出力画素が入力画素に対して、どれだけずれているかを示す位相から計算される。例えば、図11に示す2:3の拡大画素数変換の場合、上記 Q_j の出力画素の位相はその近傍の入力画素(例えば R_i の入力画素)の位相と一致しているのでその位相情報 P_1 はゼロとなり、同様に上記 Q_{j+1} の出力画素の位相はその近傍の入力画素(例

$$\begin{aligned} Q_j &= \text{Cub}(-1) * R_{i-1} + \text{Cub}(0) * R_i + \text{Cub}(1) * R_{i+1} + \text{Cub}(2) * R_{i+2} \\ Q_{j+1} &= \text{Cub}(-5/3) * R_{i-1} + \text{Cub}(-2/3) * R_i + \text{Cub}(1/3) * R_{i+1} + \text{Cub}(4/3) * R_{i+2} \\ Q_{j+2} &= \text{Cub}(-4/3) * R_i + \text{Cub}(-1/3) * R_{i+1} + \text{Cub}(2/3) * R_{i+2} + \text{Cub}(5/3) * R_{i+3} \\ &\dots (3) \end{aligned}$$

上記 $\text{Cub}(x)$ 及び入力画素の各値 R_{i-1} , R_i , R_{i+1} , R_{i+2} はそれぞれ既知の値であるので、この式(3)から各出力画素の補間データが計算できる。例えば、上記☆

$$Q_j = 0 * R_{i-1} + 1 * R_i + 0 * R_{i+1} + 0 * R_{i+2} = R_i \quad \dots (4)$$

となり、入力画素の値そのものとなる。

【0015】以上、2:3拡大画素数変換の場合を例にとって説明したが、任意の拡大比率でも同様であり、出力画素の位相さえわかれば、その位相によって式(1)からキュービック関数の各係数を求め、補間画素近傍の入力画素4点と畳み込み演算を行えばよい。

【0016】次に、例えば入力画素3個に対して出力画

素3個を作り出すようにしており、入力画素と出力画素の関係は、出力画素の値がその近傍の入力画素から計算されるという関係になっている。上記出力画素を生成するための上記近傍範囲としてどのくらいまでの範囲を使用するか、或いは入力画素から補間により出力画素を計算する際の各係数の値としてどのような係数値を使用するかなどにより、様々な補間法が存在するが、以下の説明では、上記近傍範囲として4点(4画素)分の範囲から補間するキュービック補間を例に挙げている。

【0012】上記キュービック補間にて使用されるキュービック補間関数 $\text{Cub}(x)$ を図12に示し、その関数式を式(1)に示す。ただし、式(1)に示されるキュービック補間関数の横軸は原画像をデジタル信号にサンプリングする際のサンプリング間隔で正規化されているものとする。

【0013】

※算で表され、出力画素の補間値は次式(2)のように表すことができる。

★例えば R_i の入力画素)の位相から2/3ずれているのでその位相情報 P_2 は2/3となり、上記 Q_{j+2} の出力画素の位相はその近傍の入力画素(例えば R_{i+1} の入力画素)の位相から1/3ずれているのでその位相情報 P_3 は1/3となるので、上記式(2)は式(3)のように書き換えることができる。

【0014】

☆ Q_j の出力画素に限って言えば、前記式(1)より、 $\text{Cub}(-1)=0$, $\text{Cub}(0)=1$, $\text{Cub}(1)=0$, $\text{Cub}(2)=0$ なので、

素2個を作り出すような3:2縮小画素数変換の原理について説明する。

【0017】図13には上記3:2縮小画素数変換の原理を説明するための図を示す。なお、この図13においても前記図11と同様に、各入力画素の値をそれぞれ R_{i-1} , R_i , R_{i+1} , R_{i+2} , R_{i+3} , ...とし、各出力画素の値をそれぞれ Q_j , Q_{j+1} , Q_{j+2} , ...として

表している。また、図13の中の P_1 、 P_2 、 P_1 、 \dots も、入力画素と出力画素の位相のずれ（位相情報）を表している。

【0018】ここで、上記3：2縮小画素数変換においても、前記拡大画素数変換同様に入力画素と出力画素の関係は、出力画素の値がその近傍の入力画素から計算されるという関係になっている。この3：2縮小画素数変換

$$\begin{aligned} Q_j &= \text{Cub}(x_{11}) * R_{i-1} + \text{Cub}(x_{12}) * R_i + \text{Cub}(x_{13}) * R_{i+1} + \text{Cub}(x_{14}) * R_{i+2} \\ Q_{j+1} &= \text{Cub}(x_{21}) * R_i + \text{Cub}(x_{22}) * R_{i+1} + \text{Cub}(x_{23}) * R_{i+2} + \text{Cub}(x_{24}) * R_{i+3} \\ &\dots (5) \end{aligned}$$

当該縮小画素変換においても、上記式(5)の各係数 $\text{Cub}(x)$ は前記キュービック関数から計算される値であり、これは、求めるべき出力画素が入力画素に対し、どれだけずれているかを示す位相から計算される。上記図13に示す3：2縮小画素数変換の場合、上記 Q_j の出力画素の位相はその近傍の入力画素（例えば R_i の入力画素）の位相と一致しているのでその位相情報 P_1 はぜ

$$\begin{aligned} Q_j &= \text{Cub}(-1) * R_{i-1} + \text{Cub}(0) * R_i + \text{Cub}(1) * R_{i+1} + \text{Cub}(2) * R_{i+2} \\ Q_{j+1} &= \text{Cub}(-3/2) * R_i + \text{Cub}(-1/2) * R_{i+1} + \text{Cub}(1/2) * R_{i+2} + \text{Cub}(3/2) * R_{i+3} \\ &\dots (6) \end{aligned}$$

上記 $\text{Cub}(x)$ 及び入力画素の各値 R_{i-1} 、 R_i 、 R_{i+1} 、 R_{i+2} 、 \dots はそれぞれ既知の値であるので、この式(6)から各出力画素の補間データが計算できる。例えば★

$$Q_j = 0 * R_{i-1} + 1 * R_i + 0 * R_{i+1} + 0 * R_{i+2} = R_i \quad \dots (7)$$

となり、入力画素の値そのものとなる。

【0022】以上、3：2縮小画素数変換の場合を例にとって説明したが、任意の縮小比率でも同様であり、出力画素の位相さえわかれば、その位相によって前記式(1)からキュービック関数の各係数を求め、補間画素近傍の入力4点と畳み込み演算を行えばよい。

【0023】従来は、上述したような画素数変換を、例えば図14に示すようなハードワイアードな構成で実現している。

【0024】この図14に示す構成において、直列接続されたレジスタ101～104は、それぞれ供給されたデータを1サンプル分づつ遅延するものであり、したがって、これらレジスタにより4段のシフトレジスタが構成されている。これらレジスタ101～104では、入力シフトコントロール信号 IE が“H”レベルのときに、入力端子100から供給された入力画素データを順次遅延させて、それぞれ1サンプリングシフトした画像データを出力する。一方、これらレジスタ101～104において、入力シフトコントロール信号 IE が“L”レベルの場合にはシフトせず前の値を保持する。上記各レジスタ101～104にてそれぞれシフトされて得られた各画像データは、それぞれ対応する乗算器111～114に送られる。

【0025】また、キュービック係数発生器105は、画素毎にキュービック係数 $C_1 \sim C_4$ を発生し、これらキュービック係数 $C_1 \sim C_4$ をそれぞれ対応する乗算器11

＊換でも、上述同様に出力画素（補間画素）をその近傍の入力画素4点から補間により計算するキュービック補間を例に挙げて説明する。

【0019】すなわちこの図13の縮小画素数変換の場合、各出力画素の補間値（例えば Q_j 、 Q_{j+1} ）の補間式は、以下の式(5)のようになる。

【0020】

※ロとなり、同様に上記 Q_{j+1} の出力画素の位相はその近傍の入力画素（例えば R_{i+1} の入力画素）の位相から1／2ずれているのでその位相情報 P_2 は1／2となるので、上記式(5)は式(6)のように書き換えることができる。

【0021】

★ば、上記 Q_j の出力画素に限って言えば、前記式(1)より、 $\text{Cub}(-1) = 0$ 、 $\text{Cub}(0) = 1$ 、 $\text{Cub}(1) = 0$ 、 $\text{Cub}(2) = 0$ なので、

30

1～114に対して乗算係数として供給する。したがって、これら乗算器111～114では、上記キュービック係数発生器105で発生したキュービック係数と、上記各シフトレジスタ101～104にてそれぞれシフトされた入力画素データとをかけ算する。この乗算器111～114の乗算結果は、加算器107により加算され、FIFO（先入れ先出し）メモリ108に入力される。

40

【0026】当該FIFOメモリ108は、縮小画素数変換処理の場合に画素データを飛び飛びに出力するために設けられているものであり、当該縮小画素数変換の場合にコントローラ106から供給されるスキップコントロール信号 SC に基づいて飛び飛びに画素データをスキップして、出力端子109に出力する。なお、FIFOメモリ108は、拡大画素数変換処理の場合には単なるFIFOメモリとして用い、単なるディレイ素子ではない。

【0027】コントローラ106は、拡大或いは縮小画素数変換を行う際の変換比率に基づいて、出力ポートメモリである上記FIFOメモリ108のスキップコントロール信号 SC 及びシフトレジスタ101～104の入力シフトコントロール信号 IE の生成、さらにキュービック係数発生器105のためのタイミングコントロールを行うものである。

【0028】図15は上記図14のハードウェア構成における2：3拡大画素数変換処理時の画素配置とキュー

ピック係数 C_1, C_2, C_3, C_4 との関係を示しており、当該2:3拡大画素数変換処理を行う場合にはこの図15に示すように、上記入力シフトコントロール信号IEによって3画素分入力画素データをシフトし、1画素前の画素データをシフトしないという操作を繰り返す。図14の各乗算器111~114への入力データ $D_1, *$

$$Q = C_1 * D_1 + C_2 * D_2 + C_3 * D_3 + C_4 * D_4 \quad \dots (8)$$

なお、ここでは簡単のため、2:3拡大画素数変換の例を示したが、任意の拡大比率の場合は、タイミング制御が異なるだけで原理は同じであるので、それらの説明については割愛する。

【0030】また、図16には上記図14のハードウェア構成における3:2縮小画素数変換処理時の画素配置とキュービック係数 C_1, C_2, C_3, C_4 との関係を示している。なお、図中Skipはスキップされる出力画素を示している。当該縮小画素数変換処理の場合には、前記拡大画素数変換の時と異なり、上記入力シフトコントロール信号IEは常時“L”レベルとなされ、入力画素データは各レジスタ101~104にそのまま入ってくるため、各乗算器111~114の入力データ $D_1 \sim D_4$ は図16の乗算器入力 $D_1 \sim D_4$ のようになり、これとキュービック係数 $C_1 \sim C_4$ の畳み込み演算の式(8)を行うことで所望の結果が得られる。ただし、当該3:2縮小画素数変換の場合には、出力される3画素に対して、入力の1画素が不要になるので、当該不要な画素は前記FIFOメモリ108に対する書き込みをコントロールすることによってスキップする。このための制御信号が図16に示すような出力画素のスキップコントロール信号SCとなる。すなわち、このスキップコントロール信号SCは、“H”レベルのときスキップし、“L”レベルのときスキップしない、というようにFIFOメモリ108を制御するための信号である。

【0031】なお、ここでは簡単のため、3:2縮小画素数変換の例を示したが任意の縮小比率の場合、そのタイミング制御が異なるだけで原理は同じであるのでここではそれらについての説明は割愛する。

【0032】このように、画素数変換或いは走査線数変換は、従来より上述したようないわゆるASIC(Application Specific Integrated Circuit: 特定用途向けIC)等の高速積和回路を用いて実現されている。

【0033】

【発明が解決しようとする課題】しかし、上述のような様々なフォーマットに対応するため、さらには近年のように各種の新たなフォーマットが提案されてくる状況では、上記ASICの場合、その回路規模の点、或いは設計後のビット精度の変更、上記新たなフォーマットの仕様の追加等のフレキシビリティに弱いため、市場のマーケットニーズに合わせて製品化することが困難となっている。すなわち、ASICで画素数変換等を実現するには、どうしても自由度の少ない、ある固定された変換比

* D_2, D_3, D_4 は、この図15の乗算器入力 D_1, D_2, D_3, D_4 のようになり、式(8)に示すように、これら乗算器入力とキュービック係数 C_1, C_2, C_3, C_4 との畳み込み演算を行うことで所望の結果が得られる。

【0029】

率となるか、或いは多くても数種類程度の変換比率を切り替えて使用するというような方式に限定せざるを得ない。また、上記ASICにおいては、一度回路を作成した後は、ビット精度を変更することが容易ではなく、さらに前記VGA, SVGA, 或いはXVGA, HDTV等の様々な信号フォーマットのみならず、今後出てくるであろう新たなフォーマットも含めた各種のフォーマットに全て対応させることは事実上不可能である。

【0034】そこで、本発明はこのような状況に鑑みてなされたものであり、任意比率の画素数変換や走査線数変換のためのデジタル信号処理を実現可能にし、さらには、1走査線上で水平位置により変換比率が異なった画素数変換や高品位テレビジョン等にも柔軟に対応できるようにすると共に、設計後のビット精度の変更、或いは新たなフォーマットの仕様の追加等にも柔軟に対応できる画像信号処理装置を提供することを目的とする。

【0035】

【課題を解決するための手段】本発明は、デジタル化された2次元画像の1次元方向の各画素に対応して配置すると共に1次元方向の各画素データが時系列に順次入力する複数の要素プロセッサと、各要素プロセッサを共通に制御するための制御手段とを備える画像信号処理装置であって、各要素プロセッサは、画素データを一時的に保存する一時保存手段と、入力画素データを格納して一時保存手段に転送する入力画素データ格納手段と、画素の属性を表す画素属性情報を格納する画素属性情報格納手段と、画素データをスキップさせる画素スキップ情報を格納する画素スキップ情報格納手段と、画素属性情報に基づいて入力画素データ又は近傍の要素プロセッサの画素データを用いた所定の演算を行う算術演算手段と、一時保存手段から取り出された入力画素データ或いは演算後の画素データを格納して出力する出力画素データ格納手段とを有してなることにより、上述した課題を解決する。

【0036】ここで、本発明の画像信号処理装置において、例えば画素数を拡大する場合には、画素スキップ情報に基づいて入力画素データ格納手段が入力画素データを離散的に格納し、出力画素データ格納手段が一時記憶手段からの画素データを連続的に格納する。逆に画素数を縮小する場合には、画素スキップ情報に基づいて入力画素データ格納手段が入力画素データを連続的に格納し、出力画素データ格納手段が一時記憶手段からの画素データを離散的に格納する。また、本発明装置では、デ

ータを出力するためのレートと、データ入力のレートとは独立に制御ことで、サンプリング周波数変換をも行う。画素属性情報は、各要素プロセッサ内部で生成することもできるが、要素プロセッサの外部に設けた画素属性情報生成手段にて生成することもできる。さらに、要素プロセッサの外部には画素属性情報生成手段と共に、画素スキップ情報を生成する画素スキップ情報生成手段をも設けることができ、この場合、画素スキップ情報及び上記画素属性情報と、入力画素データとを切り換えて各要素プロセッサに入力するための入力切換手段や、この入力切換手段を所定のタイミングで切換制御する切換制御手段を設けることもできる。このような要素プロセッサの外部に設ける画素スキップ情報生成手段及び画素属性情報生成手段は、順序演算、例えばモジュロ演算、四則演算、論理演算等を行う構成にて実現できる。また、所定のタイミングとしては、プランキング期間の任意の1走査線時間に、画素スキップ情報及び画素属性情報を各要素プロセッサに切り換え入力するための切換タイミングを挙げることができる。

【0037】すなわち、本発明によれば、任意比率の画素数変換処理をSIMD制御のリニアアレイ型多並列プロセッサを使い、ソフトウェア処理だけで実現可能としている。また、画素スキップ情報生成手段と画素属性情報生成手段とを備え、これら情報を所定のタイミングで入力画素データと切り換えて各要素プロセッサに供給することで、余分なプログラムエリアとワーキングメモリエリアを消費することなく、画素数変換や1走査線上で水平位置により変換比率が異なった画素数変換等を可能にしている。

【0038】

【発明の実施の形態】以下、本発明の好ましい実施の形態について、図面を参照しながら説明する。

【0039】本発明実施例の画像信号処理装置では、前述したハードウェア構成による画素数変換或いは走査線数変換のためのデジタル信号処理の問題点を打破するため、前記ASICのようなハードワイアード構成ではなく、DSP(digital signal processor)を用いたソフトウェアプログラムにて、当該デジタル信号処理を実現するようにしている。

【0040】このように、デジタル信号処理をソフトウェアプログラムにて行うことにより、仕様変更にも柔軟に対応でき、ソフトウェアプログラムを書き換えるだけで様々な異なる信号処理を切り替えて実行することを可能にしている。また、仕様の変更に対しても、ハードウェアは一切変更する必要がないので、いわゆるTAT(Time-Axis Transform System)期間を従来に比べかなり短縮することが可能となる。

【0041】前述した例えば画素数変換処理をソフトウェアプログラムにて実現するためのDSPとして、例えばいわゆるリニアアレイ型多並列プロセッサの基本的内

部構成及び基本動作を、以下に説明する。

【0042】上記リニアアレイ型多並列プロセッサとは、例えば図1に示すように、入力画素の1画素に相当する要素プロセッサ40を一次元に1走査線分並べ、これら1走査線毎に並列処理することを特徴とするものである。

【0043】この図1において、入力端子30に供給された時系列の入力画素データであるシリアル入力データSIDは、各要素プロセッサ40の入力レジスタ41に10 入力された後、データを一時的に保存するためのローカルメモリ43に転送される。また、上記ローカルメモリ43のメモリアドレスを発生するメモリアドレスジェネレータ31とインストラクションジェネレータ32は、全ての要素プロセッサ40に対して共通な制御、すなわちいわゆるSIMD(Single Instruction Multiple Data Stream)制御を行う。

【0044】このように、リニアアレイ型多並列プロセッサの特徴でもあるが、一旦DSP内部に取り込まれたデータは、1走査線分に相当する全ての要素プロセッサ40について同じ処理が施される。具体的に言うと、各要素プロセッサ40のローカルメモリ43に転送されたデータは、それぞれ演算処理部44との間で前記補間に必要な演算が施された後、出力レジスタ42に送り込まれ、最終的にこれら各要素プロセッサ40の各出力レジスタ42から出力されることで、当該DSPからは1走査線分の補間画素データが出力画素データ(シリアル出力データSOD)として取り出されることになる。

【0045】また、各要素プロセッサ40の一つ一つは、上述したように1走査線の各画素に対応しており、30 各要素プロセッサ40はそれぞれ左右近傍の他の要素プロセッサ40のローカルメモリ43内のデータにアクセス可能な構造になっている。このような構造を有することで、当該DSPでは、各要素プロセッサ40のローカルメモリ43に書き込まれた全体で1走査線分の画素データに対して、その左右近傍のデータをロードでき、これらデータをそれぞれの演算処理部44との間で送受して演算することにより、いわゆる水平方向のFIRフィルタ(非巡回型フィルタ)を実現できるようになっている。

【0046】また、当該DSPでは、全体として1走査線分の画素データを各要素プロセッサ40のローカルメモリ43に離散的に記憶でき、したがって、例えば入力時に要素プロセッサ40の1個おきに画素データを記憶することも可能である。同様に、当該DSPでは、各要素プロセッサ40の各ローカルメモリ43に記憶した全体として1走査線分に相当する画素データを、離散的に出力することも可能である。

【0047】ここで、上述したようなSIMD制御がなされるリニアアレイ型多並列プロセッサにおいて、例えば変換比率が2倍以上或いは1/2倍以下等の任意の変

換比率の画素数変換処理を簡易に実現する手法を、図2以降の各図を用いて説明する。なお、この図2の構成は、基本的には図1と同様にSIMD制御されるものであるが、簡略化のために図2には主要部のみを示している。

【0048】この図2に示す本発明実施例のリニアアレイ型多並列プロセッサ1の各要素プロセッサ10は、前記同様の入力レジスタ11及び出力レジスタ13と、後述する入力スキップレジスタ12及び出力スキップレジスタ14と、ワーキングエリアを有するローカルメモリ15と、当該ローカルメモリ15又は近傍の別の要素プロセッサ10のローカルメモリ15との間でデータの送受を行って必要な演算（フィルタ演算）を行う演算処理部16とを主要構成要素として有してなるものである。それぞれの各要素プロセッサ10は入力画素の1画素に相当し、これら各要素プロセッサ10が一次的に1走査線分並べられており、当該リニアアレイ型多並列プロセッサ1では当該1走査線毎の各要素プロセッサ10を並列処理する。

【0049】この図2に示すプロセッサ1の内部構成は、前述した図1と略々同じであるため、ここではそれぞれの機能についての説明は割愛するが、当該図2の構成は、前記図1の構成に対し、入力スキップレジスタ12と出力スキップレジスタ14の2つを明示している。以下にこれらの入力スキップレジスタ12及び出力スキップレジスタ14の動作について、他の構成要素と絡めて説明する。

【0050】この図2に示すリニアアレイ型多並列プロセッサ1において、各要素プロセッサ10では、入力された1走査線分の入力画素データを離散的或いは連続的に格納でき、また、1走査線分に相当する画素データを離散的或いは連続的に出力することを可能にしている。

【0051】ここで、当該リニアアレイ型多並列プロセッサ1の各要素プロセッサ10において、上記離散的すなわち飛び飛びに画素データを入力或いは出力させるためには、例えば”1”でスキップ、”0”でスキップしないというような意味を持たせた画素スキップ情報を、各要素プロセッサ10の入力或いは出力側に割り当てれば良い。本実施例のプロセッサ1においては、このような1ビットからなる画素スキップ情報を上記各要素プロセッサ10の入力と出力側に割り当てるために、各要素プロセッサ10に対して上記1ビットの画素スキップ情報を格納する格納手段として、入力側に上記入力スキップレジスタ12を設け、出力側に上記出力スキップレジスタ14を設けるようにしている。これら入力スキップレジスタ12、出力スキップレジスタ14に対して、上記画素スキップ情報を予め格納しておけば、後述するように各要素プロセッサ10における入力時または出力時に、画素をスキップするかスキップしないかを設定することができる。すなわち、各要素プロセッサ10では、

入力スキップレジスタ12、出力スキップレジスタ14に格納された画素スキップ情報を参照することで、入力された画素データ或いは出力する画素データをスキップするかスキップしないかを決定することができる。

【0052】より具体的に説明すると、上記入力スキップレジスタ12に格納された画素スキップ情報は拡大画素数変換処理時に画素データをスキップするための情報であり、当該拡大画素数変換の際に、各要素プロセッサ10の入力レジスタ11では、供給された入力画素データを上記画素スキップ情報に基づいて離散的に格納し、当該格納した入力画素データをローカルメモリ15に転送する。すなわち例えば、上記入力レジスタ11は画素スキップ情報が例えば”0”のときは入力画素データを格納し、画素スキップ情報が例えば”1”のときは入力画素データを格納しない（スキップする、或いは後述するように値が特定されないダミーデータを格納する）。なお、当該拡大画素数変換時の出力スキップレジスタ14に格納される画素スキップ情報は全て”0”となり、したがって出力レジスタ13は画素データをそのまま出力、すなわち連続的に出力する。

【0053】一方、上記出力スキップレジスタ14に格納された画素スキップ情報は縮小画素数変換処理時に画素データをスキップするための情報であり、当該縮小画素数変換の際に、各要素プロセッサ10の出力レジスタ13では、上記ローカルメモリ15上から読み出された画素データを、当該画素スキップ情報に基づいて離散的に格納して出力する。すなわち例えば、上記出力レジスタ13は画素スキップ情報が例えば”0”のときは画素データを格納し、画素スキップ情報が例えば”1”のときは画素データを格納しない（スキップする）。なお、当該縮小画素数変換時の入力スキップレジスタ12に格納される画素スキップ情報は全て”0”となり、したがってこの場合の入力レジスタ11は入力画素データをそのまま、すなわち連続的にローカルメモリ15に転送する。

【0054】また、本実施例のリニアアレイ型多並列プロセッサ1の各要素プロセッサ10においては、ローカルメモリ15にワーキングエリアとしての領域を設けており、このワーキングエリアを使用して、例えば演算処理部16との間でデータの送受を行うことで後述するような画素の属性を表す位相情報番号の生成やキュービック係数の選択を行い、当該生成した位相情報番号を上記当該ローカルメモリ15内に格納する。以下、この位相情報番号が格納されるローカルメモリ15内の領域を位相情報番号格納レジスタと呼び、キュービック係数が格納されるローカルメモリ15内の領域をキュービック係数格納レジスタと呼ぶことにする。

【0055】各要素プロセッサ10は、上述したようにその一つ一つが1走査線の各画素に対応しており、各要素プロセッサ10ではそれぞれ左右近傍の要素プロセッサ

サ10のローカルメモリ15に格納しているデータをアクセス可能な構造になっている。この機構により、当該リニアアレイ型多並列プロセッサ1においては、全要素プロセッサ10が同時にその左右近傍の他の要素プロセッサ10のローカルメモリ15に格納されている画素データや位相情報番号、キュービック係数等をロードでき、したがって、各要素プロセッサ10の演算処理部16ではこれらデータを使用して例えばFIRフィルタ演算を実現できる。

【0056】上記演算処理部16でのフィルタ演算により生成された補間画素データは、再びローカルメモリ15内に格納される。

【0057】ただし、リニアアレイ型多並列プロセッサの特徴でもあるSIMD制御により、1走査線分に相当する全ての要素プロセッサ10では、同じ処理が行われる。すなわち、図2では図示を省略しているが、上記演算処理部16の命令コードを生成する前記インストラクションジェネレータ32と、ローカルメモリ15のアドレスデータを生成するメモリアドレスジェネレータ31は、全ての要素プロセッサ10に対して共通な制御を行う。なお、上記左右近傍の他の要素プロセッサ10へのアクセスも、上記SIMD制御より全要素プロセッサ10共通動作であり、同時に各要素プロセッサ10毎に異なるアクセスはできない。

【0058】上記ローカルメモリ15に格納された上記画素データは、出力レジスタ13に送り込まれ、最終的に出力レジスタ13からシリアル出力データSODとして1走査線分のデータが出力されることになる。

【0059】縮小画素数変換の場合には、出力スキップレジスタ14に格納されている画素スキップ情報に基づいて、上記ローカルメモリ15から読み出された画素データが出力レジスタ13に離散的に格納され、この格納された画素データが出力レジスタ13から出力されることになる。

【0060】次に、前述したキュービック補間関数を使った画素数変換を、上記図2に示したリニアアレイ型多並列プロセッサ1にて実現する方法について説明する。なお、画素数変換では、画素数の拡大と縮小の他に等倍（すなわち1：1）変換もあるが、これは拡大、縮小の境界条件であり、どちらかに含めることができるので、ここでは等倍変換については拡大に含めることにする。

【0061】先ず、拡大画素数変換の例から説明する。

【0062】ここでは、具体的な変換比率として2：3拡大画素数変換を例に挙げ、図3を用いて詳細に説明する。任意の比率でも基本的には考え方は同じである。画素数変換は、補間画素近傍の4点の入力画素データとキュービック補間関数の畳み込み演算であることは先に述べたので、ここではSIMD制御下でどのように畳み込み演算を行うかについて示す。なお、図3には、ローカルメモリ15の位相情報格納レジスタに格納される前記

位相情報番号と、入力スキップレジスタ12及び出力スキップレジスタ14にそれぞれ格納される画素スキップ情報と、上記ローカルメモリ15内に格納される各画素データとの関係を示している。また、この図3の例では、一つの要素プロセッサ10が当該図3の縦方向の1列と対応している。また、図中Y及びQはそれぞれ画素データを示し、d_{IN}、d_{L1}、d_C、d_{R1}、d_{R2}はローカルメモリ15上にそれぞれ別々に画素データを格納する格納領域（レジスタ）を示しており、レジスタd_{L1}、d_C、d_{R1}、d_{R2}には上記位相情報番号に応じた近傍4点の画素データが格納されることになる。なお、レジスタd_{IN}は入力画素データが格納される。

【0063】上記2：3拡大画素数変換では、出力画素数が入力画素数に対して3／2倍になされるため、データ入力時には2画素おきに1画素の割合でダミーデータMを入れて、入力の段階でトータルの画素数を出力の画素数と等しくすることを行う。図3には入力される画素データの2画素おきに1画素のダミーデータMが挿入された様子を示している。上記ダミーデータMは、入力時に予め画素数を増やし、SIMD制御に適したデータ配列にするためのもので、実際の畳み込み演算ではこのダミーデータMは使われない。したがって、このダミーデータMの値は何であっても構わない。このダミーデータMを挿入するかしないか、言い換えれば入力画素データのスキップを行うか否かの設定は、1ビットあれば充分である。すなわち例えば、“1”でダミーデータMを入れる（入力画素データはスキップする）、“0”でダミーデータMを入れない（入力画素データはスキップしない）ことにすればよい。このようなダミーデータMを入れるか否かを示す情報が前記画素スキップ情報である。上記2：3拡大画素数変換を実現するには、前記入力スキップレジスタ12に対して図3のような2ビットおきに“1”が入る周期信号（画素スキップ情報）を与えて格納しておけば良い。

【0064】また、2：3拡大画素数変換では、補間演算で用いるキュービック係数も3画素周期のパターンになる。各キュービック係数そのものは、変換比率さえわかれば前記キュービック係数の計算式（1）から求められ、当該求めたキュービック係数は例えばプログラミング時にローカルメモリ15上に格納（前記キュービック係数格納レジスタに格納）しておくようにする。2：3拡大画素数変換では、上記キュービック係数が（4パターン×3）個となる。ただし、このキュービック係数は、水平方向でみた場合、どの要素プロセッサ10のローカルメモリ15内でも同じ値となるので、例えば画素毎にフィルタ係数が異なったりいわゆるポリフェーズフィルタリングを行うような場合には、画素毎（すなわち要素プロセッサ10毎）に係数セットが切り替わるように並べ換えを行わなければならない。この方法については後述する。また、この2：3拡大画素数変換例の場合、

係数セットの種類としては3種類である。

【0065】次に、畳み込み演算であるが、この2:3拡大画素数変換の場合は3通りの演算を繰り返し行えば良く、各入力画素に対しては予めそれぞれの位相の画素データを用いて計算すべきかを決めておかなければならない。この情報が前記位相情報番号PEである。本実施例のSIMD制御リニアアレイ型多並列プロセッサ1における拡大画素数変換では、初めに全要素プロセッサ10に対して上記キュービック補間の位相点を示す位相情報番号PEを割り付けるようにする。

【0066】上記2:3画素数変換(入力画素数を3/2倍にする)の場合、前述したようにFIRフィルタの位相数は3であり、上記位相情報番号PEは例えば1, 2, 3, 1, 2, 3, ...という周期パターンとなる。実際にSIMD制御のリニアアレイ型多並列プロセッサ1にて補間フィルタ計算を行う場合は、この周期パターンを1走査線分予め保持している必要がある。本実施例のリニアアレイ型多並列プロセッサ1では、上記位相情報番号PEの周期パターンを、例えば映像信号のブランキング区間等の間に計算し、各要素プロセッサ10のローカルメモリ15内に割り当てられた位相情報格納レジスタに格納しておく。

【0067】図4には、この位相情報番号PEの周期パターンを、当該リニアアレイ型多並列プロセッサ1内で計算する場合のフローチャートを示す。なお、この図4において、図中のPEは前述したような1ライン分の位相情報番号であり、上記1, 2, 3, 1, 2, 3, ...となる位相情報番号PEは、ローカルメモリ15の位相情報格納レジスタに格納される。

【0068】ここで、図4のフローチャートの処理の前提条件として、1ライン分の画素データは、先に述べたSIMD制御されることと、1ラインの中のいずれの画素に対しても、当該画素の左右近傍の画素の値を参照することができ、さらに1ラインの両端の画素の値は常にゼロであるとする。

【0069】上記位相情報番号PEの計算手順を示す図4のフローチャートにおいて、ステップST1では、上記2ビットで表されることになる位相情報番号PEの値をゼロに初期化する。具体的には、ローカルメモリ15上で上記位相情報番号PEが格納されることになる位相情報格納レジスタの値をクリアする。

【0070】次のステップST2では、全ての要素プロセッサ10において、各要素プロセッサ10では、それぞれの要素プロセッサ10のすぐ左の要素プロセッサ10のローカルメモリ15の位相情報格納レジスタに格納されている値(すなわちすぐ左の画素に対応する位相情報番号PEの値)に1を足した値を、自己のローカルメモリ15の位相情報格納レジスタに格納する。

【0071】次のステップST3では、上記ステップST2にて求めた位相情報番号PEの値が、もしも3を越

えていなければステップST2に戻り、越えていれば次のステップST4において自己のローカルメモリ15の位相情報格納レジスタに格納されている位相情報番号PEの値から3を引く。

【0072】その後、ステップST5では、上記ステップST1からステップST4までの操作が、1ラインの画素数に対応して全て終了したか否かの判定を行い、終了していないときにはステップST1に、終了したときには処理を終える。

10 【0073】なお、この図4に示した位相情報番号付け処理は、例えば電源投入時やブランキング期間等の映像信号のない期間に計算される。

【0074】次に、上述のようにして求められた位相情報番号に基づいて、キュービック係数を選択する処理の説明を行う。この説明でも2:3拡大画素数変換を例に挙げている。

20 【0075】ここで、当該キュービック係数の選択の前に、必要とされる変換比率から、出力する各画素の位相の3種類と、それに対応するキュービック係数とを、外部の例えばCPU(中央処理装置)等で予め計算しておく。このように予め計算されたキュービック係数は、例えばプログラミング時に各要素プロセッサ10のローカルメモリ15のキュービック係数格納レジスタに格納される。なお、このプログラミング時にローカルメモリ15上のキュービック係数格納レジスタに格納されるキュービック係数は、全ての要素プロセッサ10において同じ値であり、上記畳み込み演算を行うためには、各位相情報番号に基づいて各要素プロセッサ10毎に(すなわち各画素毎に)これらの係数セットを並べ換える必要がある。

30 【0076】以下に、各要素プロセッサ10のローカルメモリ15においてキュービック係数を上記位相番号情報に基づいて並べ換えて格納する手順を、図5のフローチャートを用いて説明する。

【0077】先ず、ステップST11ではローカルメモリ15上にキュービック係数C1~C4を格納するレジスタ(キュービック係数格納レジスタ)を確保する。

40 【0078】次のステップST12では、ローカルメモリ15上のキュービック係数格納レジスタに位相情報番号PE=1の係数セットをセットする。

【0079】ステップST13では、当該要素プロセッサ10のローカルメモリ15が前記位相情報番号格納レジスタに格納している位相情報番号PEの値が、2となっているか否かの判断を行い、当該位相情報番号PEの値が2となっているならばステップST14にて当該キュービック係数格納レジスタに位相情報番号PE=2の係数セットをセットし、位相情報番号PEの値が2となっていないのならば前の係数セットのままにする。

50 【0080】次に、ステップST15では、当該要素プロセッサ10のローカルメモリ15が格納している位相

情報番号 P E の値が 3 となっているか否かの判断を行い、位相情報番号 P E の値が 3 となっているならばステップ S T 1 6 にて当該キュービック係数格納レジスタに位相情報番号 P E = 3 の係数セットをセットし、位相情報番号 P E の値が 3 となっていないのならば前の係数セットのままにする。

【0081】次に、上述の位相情報番号に従って畳み込み演算するための前記 4 つの近傍画素データをローカルメモリ 1 5 にロードする処理の流れを、図 6 のフローチャートを用いて説明する。この図 6 のフローチャートでも前記 2 : 3 拡大画素数変換の場合を例に挙げている。

【0082】この図 6 のフローチャートにおいて、ステップ S T 2 1 では各要素プロセッサ 1 0 のローカルメモリ 1 5 上に上記レジスタ d L 1 , d c , d R 1 , d R 2 を確保する。

【0083】次のステップ S T 2 2 では、レジスタ d L 1 にレジスタ d i n の画素データすなわち入力画素の 2 つの左隣の画素データ（2 つ左隣の要素プロセッサ 1 0 の入力画素データ）を格納する。

【0084】次のステップ S T 2 3 では、このときの位相情報番号 P E が 2 か否かを判定し、位相情報番号 P E が 2 ならばステップ S T 2 4 に、2 でないならばステップ S T 2 6 に進む。

【0085】ステップ S T 2 3 にて位相情報番号 P E が 2 であると判定されて進むステップ S T 2 4 では、レジスタ d L 1 にレジスタ d i n の画素データすなわち入力画素の 3 つの左隣の画素データ（3 つ左隣の要素プロセッサ 1 0 の入力画素データ）を格納する。その後、ステップ S T 2 5 では、レジスタ d c にレジスタ d i n の画素データ（入力画素データ）を格納し、ステップ S T 2 6 に進む。

【0086】上記ステップ S T 2 6 では、そのときの位相情報番号 P E が 3 か否かを判定し、位相情報番号 P E が 3 ならばステップ S T 2 7 に、3 でないならばステップ S T 2 9 に進む。

【0087】ステップ S T 2 6 にて位相情報番号 P E が 3 であると判定されて進むステップ S T 2 7 では、レジスタ d c にレジスタ d i n の画素データすなわち入力画素の 1 つの左隣の画素データ（1 つ左隣の要素プロセッサ 1 0 の入力画素データ）を格納する。その後、ステップ S T 2 8 では、レジスタ d R 1 にレジスタ d i n の画素データの 1 つ右隣の画素データ（1 つ右隣の要素プロセッサ 1 0 の入力画素データ）を格納し、ステップ S T 2 9 に進む。

【0088】上記ステップ S T 2 9 では、そのときの位相情報番号 P E が 1 か否かを判定し、位相情報番号 P E が 1 ならばステップ S T 3 0 に、1 でないならばステップ S T 3 2 に進む。

【0089】ステップ S T 2 9 にて位相情報番号 P E が 1 であると判定されて進むステップ S T 3 0 では、レジ

スタ d R 1 にレジスタ d i n の画素データすなわち入力画素データを格納する。その後、ステップ S T 3 1 では、レジスタ d R 2 にレジスタ d i n の画素データの 3 つ右隣の画素データ（3 つ右隣の要素プロセッサ 1 0 の入力画素データ）を格納し、ステップ S T 3 2 に進む。

【0090】上記ステップ S T 3 2 では、そのときの位相情報番号 P E が 2 か又は 3 であるか否かを判定し、位相情報番号 P E が 2 又は 3 であるならばステップ S T 3 3 に進む。

【0091】ステップ S T 3 2 にて位相情報番号 P E が 2 又は 3 であると判定されて進むステップ S T 3 3 では、レジスタ d R 1 にレジスタ d i n の画素データすなわち入力画素の 2 つ右隣の画素データ（2 つ右隣の要素プロセッサ 1 0 の入力画素データ）を格納する。

【0092】その後は処理を終了する。

【0093】以上の操作により、ローカルメモリ 1 5 上にはキュービック係数と 4 つの近傍画像データが得られるので、これらを用いて前記式 (8) に示した畳み込み演算を行うことで、補間画素データが求められることになる。上述のようなローカルメモリ 1 5 上のデータ分布が前記図 3 に示されている。

【0094】次に、縮小画素数変換の場合を説明する。

【0095】ここでは、ここでは、具体的な変換比率として、3 : 2 縮小画素数変換の場合について説明する。なお、この縮小画素数変換の場合も上記 3 : 2 に限らず、任意の変換比率であっても基本的な考え方は同じである。

【0096】この縮小画素数変換の場合は、入力時に画素データはそのまま入力され、画素データを出力するときに、前記出力スキップレジスタ 1 4 に格納された画素スキップ情報に基づいて、ローカルメモリ 1 5 から読み出された画素データが離散的に（飛び飛びに）出力レジスタ 1 3 に格納されて出力されるところが、前記拡大画素数変換と異なるだけであり、基本的な手順は前記拡大画素数変換の場合と同じである。

【0097】上記 3 : 2 縮小画素数変換の場合は、出力画素数が入力画素数の 2 / 3 倍になるため、データ入力時には入力画素データをそのまま入力し、出力の際に 3 画素につき、2 画素の割合でスキップして出力する。すなわち、3 画素に 1 画素の割合で出力画素をスキップすればよく、これは本実施例のリニアアレイ型多並列プロセッサ 1 において 1 ビットの出力スキップレジスタ 1 4 を設け、これに図 7 に示すような周期信号（画素スキップ情報）を与えれば実現可能である。なお、図 7 は前記図 3 と同様に表しており、図 7 の図中 G s がスキップされた出力画素データを表している。上記画素スキップ情報は、" 1 " で出力画素データをスキップし、" 0 " で出力画素データをスキップしないものとなる。

【0098】また、この 3 : 2 縮小画素数変換の場合、補間演算に用いるキュービック係数は 2 画素周期のパタ

ーンになるが、SIMD制御のリニアアレイ型プロセッサ1で処理するために3つに1つのダミー位相 m を入れる。このため、実際には3画素周期になる。すなわち例えば、ダミー位相 m を3番目の位相(位相情報番号 $PE=3$)とすると、この3番目のダミー位相 m は実際スキップされて出力されないで、キュービック係数は何でもよく、ここでは入力スキップレジスタ12の当該3番目の位相に対する画素スキップ情報をゼロとしている。各キュービック係数そのものは変換比率さえわかれば前記キュービック係数の計算式(1)から求めることができるので、例えば外部CPU等で計算して、プログラミング時にローカルメモリ15上のキュービック係数格納レジスタに格納しておけばよい。当該3:2縮小画素数変換では、上記キュービック係数が(4パターン \times 3)個となる。ただし、このキュービック係数は、水平方向でみた場合、どの要素プロセッサ10のローカルメモリ15内でも同じ値となるので、例えば画素毎にフィルタ係数が異なったりするポリフェーズフィルタリングを行うような場合には、画素毎(すなわち要素プロセッサ10毎)に係数セットが切り替わるように並べ換えを行わなければならない。この方法については後述する。また、この3:2縮小画素数変換例の場合、係数セットの種類としては3種類である。

【0099】次に、畳み込み演算であるが、この3:2縮小画素数変換の場合は3通りの演算を繰り返し行えば良く、各入力画素に対しては予めそれぞれの位相の画素データを用いて計算すべきかを決めておかなければならない。この情報が前記位相情報番号 PE である。本実施例のSIMD制御リニアアレイ型多並列プロセッサにおける縮小画素数変換でも、初めに全要素プロセッサ10にこの位相情報番号 PE を割り付ける。上記リニアアレイ型多並列プロセッサにおいて縮小画素数変換処理を行う際にも、キュービック補間の位相点を示す位相情報番号 PE が必要となるが、これは前記拡大画素数変換の場合と全く同じ前記図4のフローチャートの手順にて求めることができる。当該縮小画素数変換の場合も、上記位相情報番号はローカルメモリ15の位相情報番号レジスタに格納される。

【0100】次に、前記図4と同様にして求めた位相情報番号に基づいて、キュービック係数を選択するが、この縮小画素数変換の場合も、当該キュービック係数の選択の前に、必要とされる変換比率から、出力する各画素の位相の3種類と、それに対応するキュービック係数を外部の例えばCPU(中央処理装置)等で予め計算しておく。当該予め計算されたキュービック係数は、例えばプログラミング時に各要素プロセッサ10のローカルメモリ15のキュービック係数格納レジスタに格納される。なお、当該縮小画素数変換の場合も、上記プログラミング時にローカルメモリ15に格納されるキュービック係数は、全ての要素プロセッサ10において同じ値で

あり、上記畳み込み演算を行うためには、各位相情報番号に基づいて各要素プロセッサ10毎に(すなわち各画素毎に)これらの係数セットを並べ換える必要がある。この並べ換えて格納する手順も、前記拡大画素数変換と同じ前記図5のフローチャートにて実現される。

【0101】次に、当該3:2縮小画素数変換において、前記位相情報番号に従って畳み込み演算するための前記4つの近傍画素データをローカルメモリ15にロードする処理の流れを、図8のフローチャートを用いて説明する。なお、図8において、 d_{in} , d_{l1} , d_c , d_{r1} , d_{r2} は前述同様のローカルメモリ15上のレジスタである。

【0102】図8において、ステップST41では、各要素プロセッサ10のローカルメモリ15上に上記レジスタ d_{l1} , d_c , d_{r1} , d_{r2} を確保する。

【0103】次のステップST42では、レジスタ d_{l1} にレジスタ d_{in} の画素データすなわち入力画素の1つの左隣の画素データ(1つ左隣の要素プロセッサ10の入力画素データ)を格納する。

【0104】次のステップST43ではレジスタ d_c にレジスタ d_{in} の画素データを格納し、ステップST44ではレジスタ d_{r1} にレジスタ d_{in} の画素データの1つ右隣の画素データ(1つ右隣の要素プロセッサ10の入力画素データ)を格納する。

【0105】ステップST45ではレジスタ d_{r2} にレジスタ d_{in} の画素データの3つ右隣の画素データ(3つ右隣の要素プロセッサ10の入力画素データ)を格納する。

【0106】その後は処理を終了する。

【0107】以上の操作により、ローカルメモリ15上にはキュービック係数と4つの近傍画像データが得られるので、これらを用いて前記式(8)に示した畳み込み演算を行うことで、補間画素データが求められる。ローカルメモリ15から読み出された補間画素データは、出力スキップレジスタ14の画素スキップ情報に従って飛び飛びに出力レジスタ13に格納されて出力されることになる。上述のようなローカルメモリ15上のデータ分布が前記図3と同様に表す図7に示されている。

【0108】なお、上述した本発明の第1の実施例のリニアアレイ型多並列プロセッサにおいて、データを出力するためのレート、入力と出力とで独立に制御すれば、画素数変換のみならずサンプリング周波数変換処理も可能となる。画素数変換処理とサンプリング周波数変換処理とは原理的には同じ技術である。また、映像信号の水平方向のサンプリング点を増減するのが画素数変換であるが、これを垂直方向に眺めれば走査線数変換となり、各画素を走査線と置き換えることで、同じ考え方を適用することができる。このことは、後述する第2の実施例においても同じである。

【0109】上述したように本発明の第1の実施例によ

れば、例えばNTSC信号やVGA信号などの異なる映像信号フォーマット間での相互フォーマット変換、すなわち任意の画素数変換処理或いは映像信号の任意倍率への拡大、縮小処理、異なる標本周波数（サンプリング周波数）への映像信号標本周波数変換処理を、SIMD制御のリニアアレイ型多並列プロセッサを用いて、ソフトウェア処理だけで実現可能としている。また、このようにシステムをソフトウェア化することで、従来のASIC等のハードウェア構成では困難であった任意比率の画素数変換は勿論のこと、設計後のビット精度の変更、或いは新たなフォーマットの仕様の追加等にも柔軟に対応できる。さらに、ハードウェアは変更する必要がなく、TAT期間を従来に比べてかなり短縮することができるので、市場のマーケットニーズに合わせた製品化が可能となる。

【0110】ところで、本発明の第1の実施例では、リニアアレイ型多並列プロセッサ内部において、前記位相情報番号の生成やキュービック係数の並べ換えを例えばブランキング期間内で行う例を挙げているが、この第1の実施例の場合、処理によっては以下のようなことが生

ずる場合がある。

【0111】（A）例えばフィールド毎に画素数を変えるような場合、前記位相情報番号付けのような一連の画素属性情報番号付作業（以下、ナンバリングと呼ぶ）は必ずブランキング期間内で終了していなければならず、高品位TV等の画素数が多く且つブランキング期間が短いような場合には、ナンバリングが終了しないことが起こり得る。

【0112】（B）ナンバリング操作のために余分にローカルメモリの容量（例えば前記ワーキングエリア）が必要となり、また当該ナンバリング操作のためのプログラムを格納するためのプログラムメモリエリアも必要となり、プロセッサの実際の信号処理に割り振られるメモリを圧迫してしまうことになる。これらメモリサイズがチップコストに直に反映され、コストアップにつながる可能性がある。

【0113】（C）SIMD制御のため規則的なパターン、つまり拡大あるいは縮小の比率が1走査線中どこでも一定なものは作りやすいが、場所により比率が異なるような場合には、上記ナンバリングはできない。

【0114】このような場合の対処法が、本発明の第2の実施例であり、以下に当該第2の実施例の構成及び動作の説明を行う。

【0115】図9には本発明の第2の実施例の概略構成を示す。なお、この図2の構成は基本的には図1と同様にSIMD制御されるものであるが、簡略化のために図9には主要部のみを示している。

【0116】この図9に示すリニアアレイ型多並列プロセッサ51は、前述同様に一画素毎に対応した要素プロセッサ60を一次的に1走査線分並べ、これら1走査

線毎に並列処理を行うようにしている。各要素プロセッサ60は、入力レジスタ61と出力レジスタ62と、ローカルメモリ63と、演算処理部65等を有している。また、この図9に示すリニアアレイ型多並列プロセッサ51は、外部付加回路として、ナンバリング回路54と入力手段55とシステム同期回路56とセクタ53とを備えている。

【0117】この図9において、入力端子52には前記時系列の入力画素データであるシリアル入力データSIDが供給される。このシリアル入力データSIDはセクタ53の被切換端子A側に送られる。当該セクタ53の被切換端子B側はナンバリング回路54の出力データが供給されるようになされている。

【0118】上記ナンバリング回路54は、前記位相情報番号と画素スキップ情報を生成して出力するものである。当該ナンバリング回路54は、具体的には図10に示すような4ビットカウンタ70と論理積（AND）回路71とを有してなるいわゆる順序回路であるが、論理積（AND）、論理和（OR）等の論理演算とカウンタと組み合わせたモジュロ回路等から構成されるものである。このナンバリング回路54は、入力手段55からのパラメータに基づいて上記位相情報番号と画素スキップ情報を生成して出力する。

【0119】上記入力手段55はパラメータ設定入力用に設けられており、当該入力手段55にて設定されるパラメータとしては、例えばカウンタの初期値やサイクル幅の設定、リピート数の設定などがある。

【0120】また、システム同期回路56は、システム同期用タイミング信号を生成する。このタイミング信号は、水平同期パルス、垂直同期パルス等の同期信号である。上記セクタ53は、システム同期回路56からのタイミング信号に応じて、ブランキング期間の任意の1走査線時間に上記被切換端子Aから被切換端子Bに切り換えられるものである。したがって、当該セクタ53からは、上記タイミング信号に応じて、上記入力端子52からの入力画素データか、又はナンバリング回路54が生成した位相情報番号及び画素スキップ情報の何れかが、出力されることになる。当該セクタ53から出力された入力画素データ、或いは位相情報番号及び画素スキップ情報は、リニアアレイ型多並列プロセッサ51の各要素プロセッサ60に入力される。すなわち、入力画素データと、上記位相情報番号及び画素スキップ情報は、別のタイミングで各要素プロセッサ60に供給されることになる。

【0121】上記位相情報番号及び画素スキップ情報は、上記各要素プロセッサ60の入力レジスタ61を介してローカルメモリ63に転送されて格納される。すなわち、第2の実施例のプロセッサ51では、位相情報番号と画素スキップ情報の両方をローカルメモリ63に記憶し、上記画素スキップ情報を格納するための前記入力

スキップレジスタや出力スキップレジスタを備えていない。なお、画素スキップ情報は前述したようにデータ量の少ないものであるため、ローカルメモリ 6 3 の容量を大きく消費してしまうことはない。

【0 1 2 2】上記ローカルメモリ 6 3 に格納された画素スキップ情報は、前述したように拡大画素数変換時に、前記入力レジスタ 6 1 に供給された入力画素データを離散的に格納させて上記ローカルメモリ 6 3 上へ転送するため、及び出力レジスタ 6 2 に供給されたローカルメモリ 6 3 からの画素データを連続的に格納して出力させるための情報であり、一方、縮小画素数変換処理時に、入力レジスタ 6 1 に狭義希有された入力画素データを連続的に格納させて上記ローカルメモリ 6 3 へ転送するため、及び上記ローカルメモリ 6 3 上から読み出された画素データを離散的に出力レジスタ 6 2 へ格納させて出力するための情報である。また、上記ローカルメモリ 6 3 に格納された位相情報番号も前述同様である。

【0 1 2 3】以下に、上述したような構成を有する第 2 の実施例のリニアアレイ型多並列プロセッサ 5 1 における画素数変換処理を説明する。ここでは、簡単のため固定比率の拡大画素数変換について説明する。

【0 1 2 4】例えば前記 2 : 3 拡大画素数変換の場合、図 1 0 のナンバリング回路 5 4 では、水平同期信号をトリガとし、4 ビットカウンタ 7 0 にて 1, 2, 3, 1, 2, 3, ... という値を出力する。この 4 ビットカウンタ 7 0 から出力される値は、前記位相情報番号である。また、2 : 3 拡大操作であるので、入力画素 3 個に対し、1 つの割合で入力画素のスキップ操作が必要であり、この画素スキップ情報はナンバリング回路 5 4 の論理積回路 7 1 により、上記位相情報番号と同時に生成されて出力される。

【0 1 2 5】このとき、上記タイミング信号によって、ブランキング期間の任意の 1 走査線時間に上記セクタ 5 3 を被切換端子 B 側に切り換えることで、上記ナンバリング回路 5 4 からの位相情報番号と画素スキップ情報からなるナンバリングデータが、当該リニアアレイ型多並列プロセッサ 5 1 の要素プロセッサ 6 0 の入力レジスタ 6 1 に送られ、さらにこれらナンバリングデータは当該要素プロセッサ 6 0 内部のローカルメモリ 6 3 に転送される。

【0 1 2 6】その後、上記セクタ 5 3 は再び被切換端子 A 側に切り換えられる。これにより、各要素プロセッサ 6 0 には通常の画像データが入力されることになる。要素プロセッサ 6 0 のローカルメモリ 6 3 内に蓄えられたナンバリングデータは、前記同様の位相情報番号、画素スキップ情報として、当該リニアアレイ型多並列プロセッサ 5 1 による FIR フィルタの係数セットの選択等に用いられる。

【0 1 2 7】なお、この第 2 の実施例においては、拡大画素数変換の場合のみ説明したが、前記縮小画素数変換

の場合も基本的には同様である。すなわち、当該縮小画素数変換の場合には、入力時に入力画素データはそのまま入力され、出力時に離散的に補間画素データが出力されるところが異なるだけで、基本的な手順は拡大と同じである。

【0 1 2 8】また、第 2 の実施例では、回路構成を簡単にするためにロジック回路にてナンバリング操作等を行っているが、勿論、CPU 等の演算手段にて実現するようにしても構わない。

【0 1 2 9】さらに、ここでは水平方向の画素数変換にしか言及していないが、垂直方向に見れば走査線数変換となり、同じことが適応できる。

【0 1 3 0】上述したように本発明の第 2 の実施例のリニアアレイ型多並列プロセッサ 5 1 によれば、前記第 1 の実施例同様に、任意比率の画素数変換処理を SIMD 制御のリニアアレイ型多並列プロセッサを使い、ソフトウェア処理だけで実現可能とし、システムをソフトウェア化することで、従来の ASIC 等のハードウェア構成では困難であった任意比率の画素数変換は勿論のこと、設計後のビット精度の変更、或いは新たなフォーマットの仕様の追加等にも柔軟に対応可能となっている。さらに、ハードウェアは変更する必要がなく、TAT 期間を従来に比べてかなり短縮することができるので、市場のマーケットニーズに合わせた製品化が可能となる。

【0 1 3 1】また、本発明の第 2 の実施例のリニアアレイ型多並列プロセッサ 5 1 においては、簡単なセクタ 5 3 とナンバリング用のハードウェア回路等を外部に備え、上記タイミング信号に応じてセクタ 5 3 を切り換え制御して、映像区間の始まる前で且つ 1 走査線期間の間に上記ナンバリング回路 5 4 からの位相情報番号と画素スキップ情報を各要素プロセッサ 6 0 内部に流し込み、これら位相情報番号と画素スキップ情報に基づいて画素数変換を行うことで、余分なプログラムエリアとワーキングメモリエリアを消費することなく、画素数変換が可能となる。また、1 走査線上で水平位置により変換比率が異なった画素数変換や高品位テレビジョン等にも柔軟に対応できる。すなわち、当該第 2 の実施例においては、フィールド毎に画素数を変えるような画素数変換処理における画素属性情報番号生成（ナンバリング）を当該プロセッサ 5 1 外部の僅かな追加回路（ロジック回路）にて行うようにしているため、高品位 TV 等の画素数が多く且つブランキング期間が短い場合のものであっても前記ナンバリングが行え、ナンバリング操作のために余分なローカルメモリ、プログラムメモリエリアが不要となるので、チップのコストアップを抑えることができ、SIMD 制御のため規則的なパターン、つまり拡大或いは縮小の比率が 1 ライン中どこでも一定でなく、場所により比率が異なるような場合でもナンバリングが可能となる。したがって、画素数が多い高品位テレビジョンの画素数変換や走査線の水平位置によって拡大、縮小

の倍率が増えるような画素数変換等も実現可能である。

【0132】

【発明の効果】本発明においては、SIMD制御のリニアアレイ型多並列プロセッサを使い、ソフトウェア処理だけで、任意比率の画素数変換や走査線数変換のためのデジタル信号処理を実現可能となっている。また、本発明においては、画素スキップ情報生成手段と画素属性情報生成手段とを備え、これら情報を所定のタイミングで入力画素データと切り換えて各要素プロセッサに供給することで、余分なプログラムエリアとワーキングメモリエリアを消費することなく、画素数変換や1走査線上で水平位置により変換比率が異なった画素数変換等が可能となっている。

【図面の簡単な説明】

【図1】リニアアレイ型多並列プロセッサの基本構成を示すブロック回路図である。

【図2】第1の実施例のリニアアレイ型多並列プロセッサの概略構成を示すブロック回路図である。

【図3】実施例の2：3拡大画素数変換の動作説明に用いる図である。

【図4】位相情報番号付けの手順を示すフローチャートである。

【図5】キュービック係数を位相情報番号により並べ換える手順を示すフローチャートである。

【図6】2：3拡大画素数変換時の近傍画素データのロード手順を示すフローチャートである。

【図7】実施例の3：2縮小画素数変換の動作説明に用

いる図である。

【図8】3：2縮小画素数変換時の近傍画素データのロード手順を示すフローチャートである。

【図9】第2の実施例のリニアアレイ型多並列プロセッサの概略構成を示すブロック回路図である。

【図10】2：3拡大画素数変換時のナンバリング回路の概略構成を示すブロック回路図である。

【図11】2：3拡大画素数変換の原理説明に用いる図である。

【図12】キュービック関数の説明に用いる図である。

【図13】3：2縮小画素数変換の原理説明に用いる図である。

【図14】従来の画素数変換装置のハードウェア構成を示すブロック回路図である。

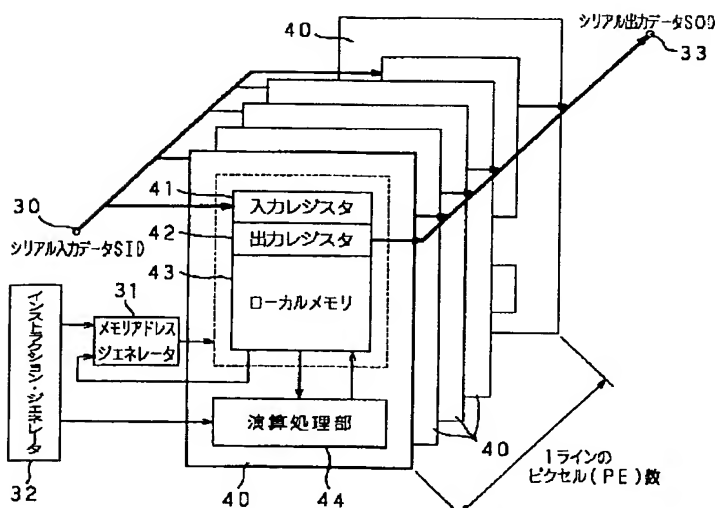
【図15】従来のハードウェア構成による画素数変換装置における2：3拡大画素数変換の動作説明に用いる図である。

【図16】従来のハードウェア構成による画素数変換装置における3：2縮小画素数変換の動作説明に用いる図である。

【符号の説明】

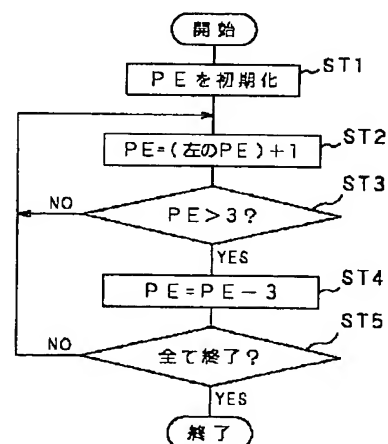
1, 51 リニアアレイ型多並列プロセッサ、10, 60 要素プロセッサ、11, 61 入力レジスタ、12 入力スキップレジスタ、13, 62 出力レジスタ、14 出力スキップレジスタ、15, 63 ローカルメモリ、16, 64 演算処理部、53 セレクタ、54 ナンバリング回路、55 入力手段、56 システム同期回路

【図1】



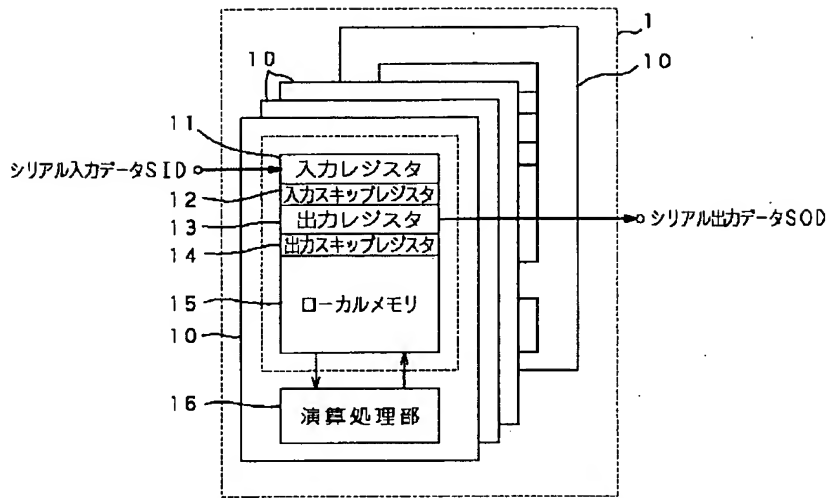
リニアアレイ型多並列プロセッサの基本構成

【図4】



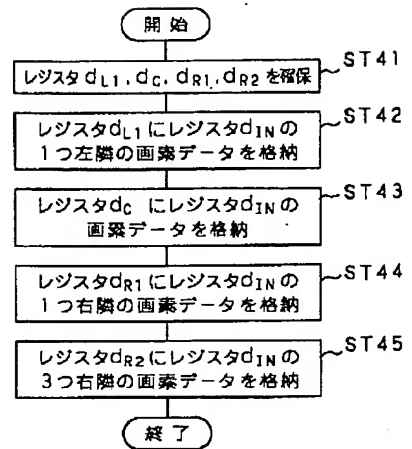
全面素に1, 2, 3, 1, 2, 3, 1, 2, 3...の番号付けを行う1方法

【図 2】



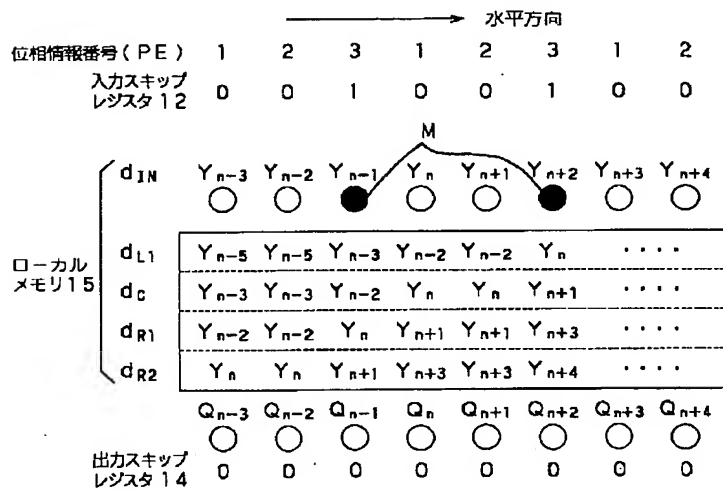
第1の実施例のリニアアレイ型多並列プロセッサの構成

【図 8】



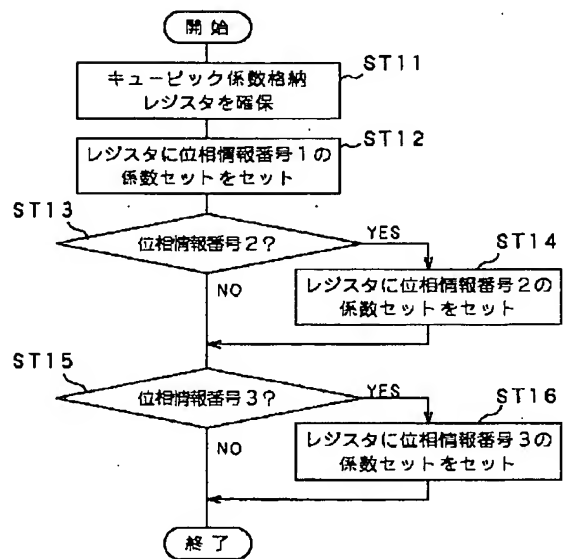
3: 2倍小画素数変換時の近傍画素データロード手順

【図 3】



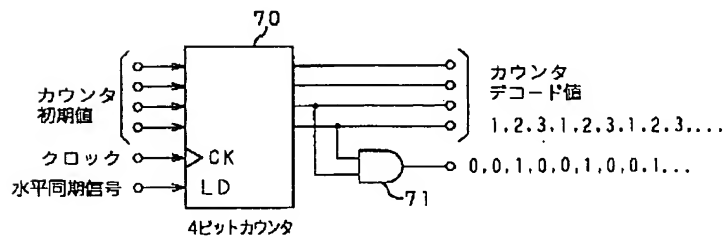
実施例の2: 3拡大画素数変換

【図 5】



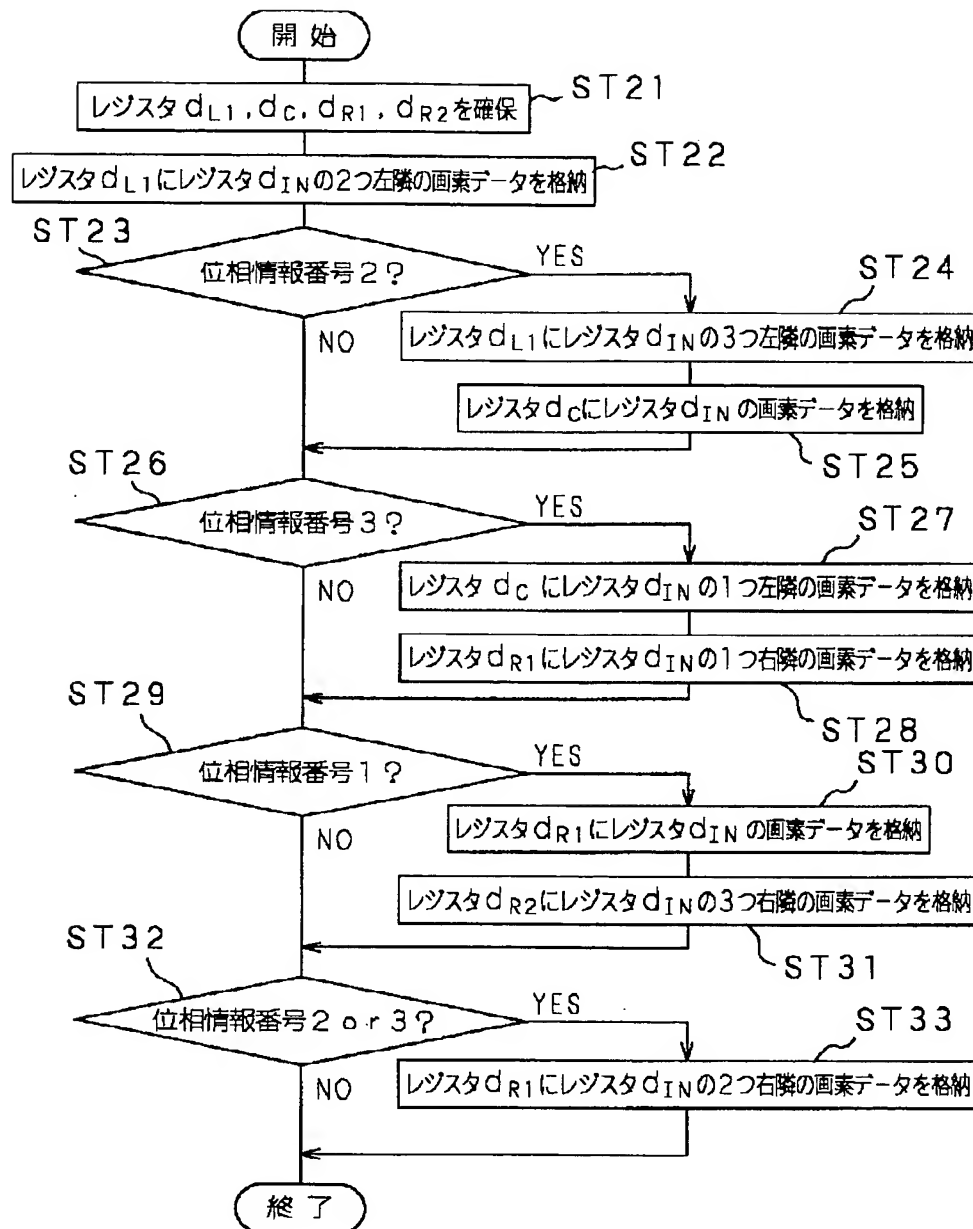
キュービック係数並べ換え手順

【図 10】



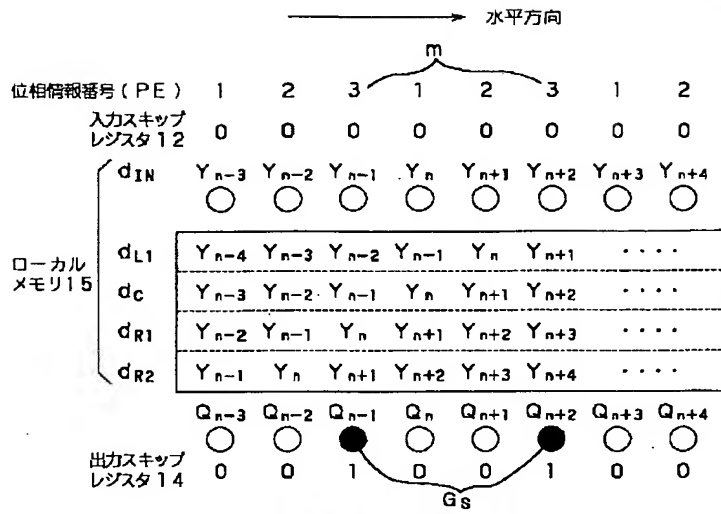
2: 3拡大画素数変換時のナンバリング回路

【図6】



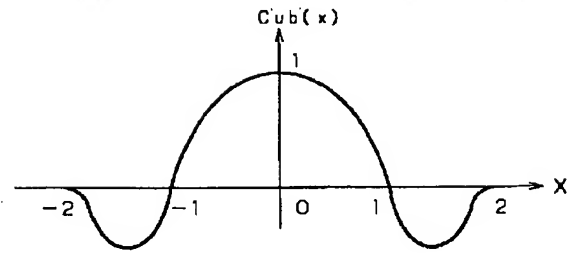
2:3拡大画素数変換時の近傍画素データのロード手順

【図7】



実施例の3: 2次元小画素変換

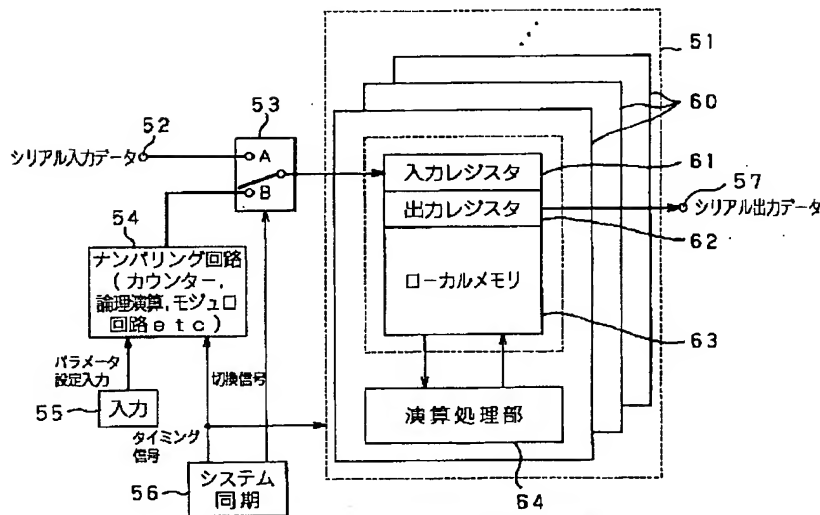
【図12】



$$Cub(x) = \begin{cases} |x|^3 - 2|x|^2 + 1 & |x| \leq 1 \\ |x|^3 + 5|x|^2 - 8|x| + 4 & 1 < |x| \leq 2 \\ 0 & 2 < |x| \end{cases}$$

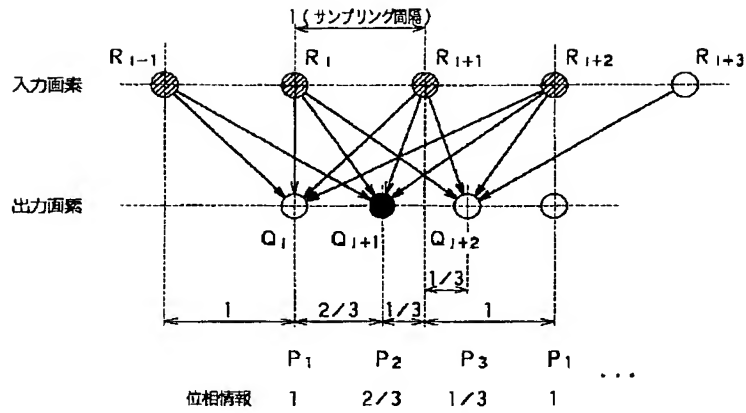
キュービック関数

【図9】



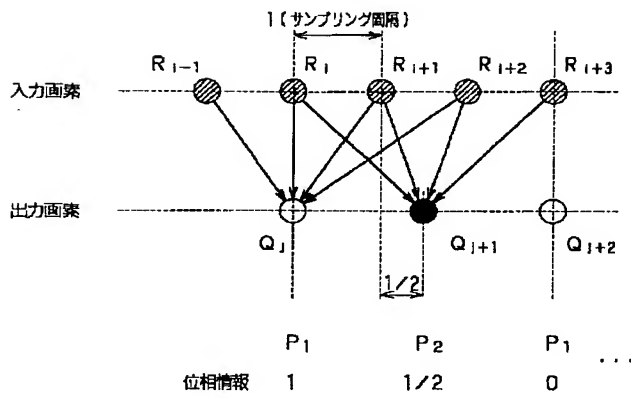
第2の実施例のリニアレイ型多並列プロセッサの構成

【図11】



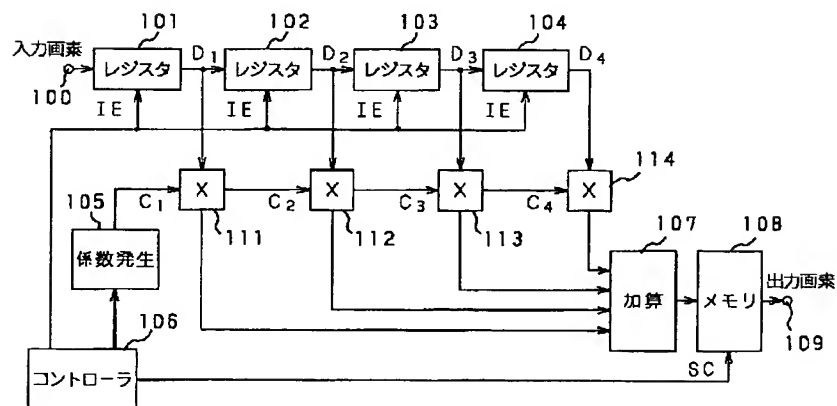
2:3画素数変換(拡大変換)

【図13】



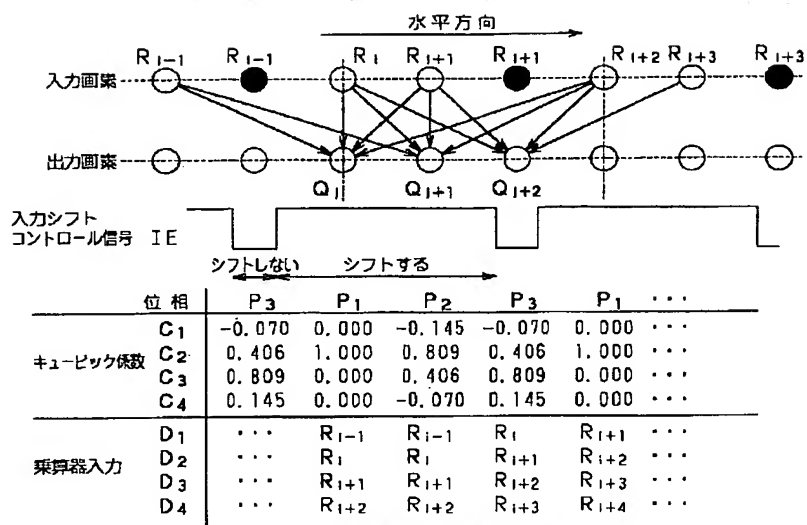
3:2画素数変換(縮小変換)原理図

【図14】



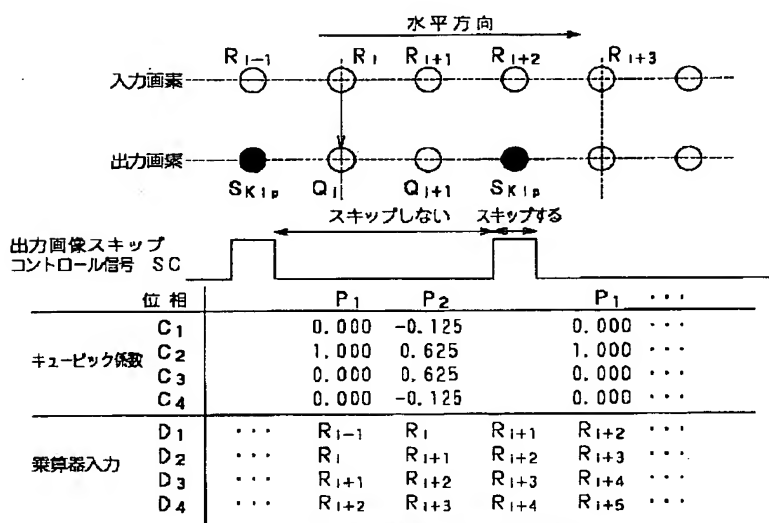
画素数変換処理のハードウェア構成図

【図15】



2:3画像数変換(拡大変換)従来例

【図16】



3:2画像数変換(縮小変換)従来例

フロントページの続き

(51) Int. Cl.⁶

H04N 7/01

識別記号

FI

G06F 15/66

355A

(72) 発明者 岩瀬 清一郎

東京都品川区北品川6丁目7番35号 ソニ
ー株式会社内

THIS PAGE BLANK (USPTO)